Plant pathlology detection with convolutional neural networks

Barış Deniz Sağlam
Informatics Institute
Middle East Technical University
Ankara, Turkey
e155841@metu.edu.tr

***Abstract*—**
***Index Terms*—Deep Learning, CNN, Plant Pathology**

I. Introduction

Apple industry in U.S. has $15 billion annual market size [1]. Various kinds of plant diseases cause significant economic losses. The manual diagnosis of apple plant diseases are laborous and expensive. Hence, computer vision-based methods have been developed to detect diseases from the images of leaves. The variations in imaging conditions such as backgrounds, lighting, and the large variety of visual symptoms are the main challenging aspects for these methods.

In this paper, we investigate the performance of convolutional neural networks, specifically, we train ResNet [2] and EfficientNet [3] architectures with different scales, on detecting plant diseases from the images of apple leaves.

A. Dataset

We use Plant Pathology 2021-FGVC8 dataset [1] for training the models. The dataset consists of over $18,632$ RGB images, varying between 2592×1728 to 5184×3456 in size, and their labels. Each sample is labelled by experts with one or more disease types due to the fact that a plant may have multiple diseases.

There are 4 kinds of plant disease existing in the dataset:

- *frog-eye leaf spot*
- *powdery mildew*
- *rust*
- *scab*

Additionally, a sample is labelled as *healthy* when no disease is spotted, and labelled as *complex* when many diseases are spotted.

As it can be seen from Fig. 2., there is imbalance among labels in the dataset.

B. Literature Search

Tan and Le [3] analyzed model scaling in convolutional neural networks and shown that when network depth, width, and resolution are balanced, the model performance improves. They proposed a new scaling method parameterized with a single coefficient and demonstrated that when the method is applied to scale existing CNNs such MobileNet, it achieves higher accuracy. To test the effectiveness of the method

Fig. 1. Sample images and labels from the dataset

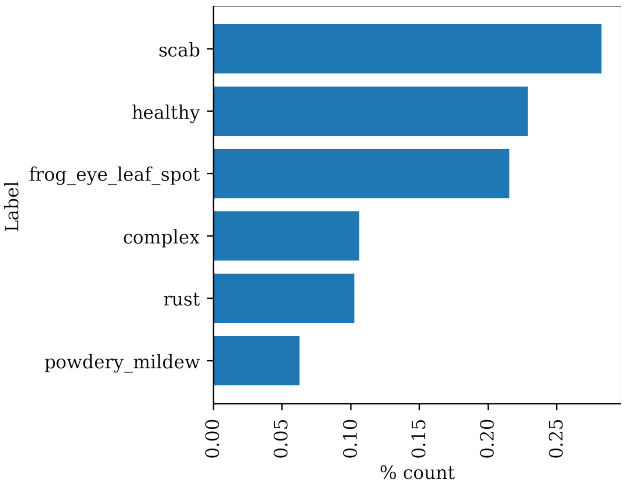

Fig. 2. Label distribution in the dataset

on a novel model architecture, first, they perform a neural architecture search for a convolutional block constrained by number of FLOPs and create full networks by scaling this block with the proposed scaling method. The family of such networks with different scales is named EfficientNets. Then, they compare the performance of EfficientNets with other well-performing convolutional neural networks such as ResNet [2], InceptionNet [4] [5] [6], and MobileNet [7] on several image classification datasets such as CIFAR [8], and ImageNet [9]. They show that EfficientNets achieve highest accuracy and faster inference on all datasets among all models with less parameters.

The paper by He et. al is a comprehensive ablation study on convolutional neural networks. It investigates the impact of training techniques such as learning rate warming, label smoothing, transfer learning, weighting loss; and hyperparameters such as batch size, learning rate on three tasks image classification, object detection, and semantic segmentation with three CNN architectures ResNet50 [2], InceptionV3 [5], and MobileNet [7]. The residual block of ResNet has convolutional layers with 1×1 kernel size and 2 stride, which ignores half of the information flowing throught the block [10]. It also proposes a few modifications to Moreover, the convolutional layers with 7×7 kernel size are computationally more expensive than several successive 3×3 convolutions. Hence, the paper proposes several tweaks to eliminate these inefficiencies in the original ResNet architecture.

Lin et al. [11] proposed a novel loss function that improves training accuracy in highly imbalanced problems such as object detection. Focal loss (Eq.1) is a modulated cross entropy loss, where the modulating factor downweighs losses for easy negative examples such as background in object detection problems. This prevents easy negatives to dominate loss function and adversely influence the training. The paper also proposes a one-stage object detection model architecture, named RetinaNet, that uses focal loss. It surpasses the accuracy of all two-stage object detection models without compromising inference speed and proves the effectiveness of focal loss. We use focal loss in our models with the hyperparameters suggested by the paper.

$$p_T = \begin{cases} p & \text{if y = 1} \\ 1-p & \text{otherwise} \end{cases}$$

$$\text{FL}(p_t) = \alpha_t (1-p_t)^\gamma \log p_t \tag{1}$$

II. Methods

A. Encoding

The targets are encoded as one-hot vectors for 5 classes (*complex*, *frog-eye leaf spot*, *powdery mildew*, *rust*, *scab*) and as zero vector for *healthy* class as it implies the absence of disease.

B. Model training and evaluation

We trained several models with ResNet, EfficientNet and XResNet architectures with different model scales. Two model input sizes have been experimented; $384 \times 384 \times 3$ and $512 \times 512 \times 3$. The dataset is split into training and validation sets with 70-30% ratios, respectively.

Since the task is multi-label classification, binary cross-entropy loss (Eq.II-B) and focal loss (Eq.1) functions have been used to train models. We trained a baseline model with ResNet18 architecture pretrained on ImageNet with no image augmentation and with binary cross entropy.

$$\text{BCE} = -(y \log(p) + (1-y) \log(1-p))$$

To improve model performance and training process, we utilized many techniques from [10] and fastai [12] library. We used learning rate finder algorithm proposed by Smith [13] and implemented in fastai library to pick an optimal learning rate for fast convergence. As learning rate scheduler, we used learning rate warmup and annealing [10] with 1cycle policy from [14]. Alongside with focal loss, we experimented with weighted binary cross entropy loss with inverse frequencies of labels to mitigate the negative effects of class imbalance on model performance. Data augmentation is a common technique in deep learning practice due to its effectivess at preventing overfitting and improving model inference performance [15]. We applied several image augmentation methods on our training set to increase generalization;

- Horizontal and vertical flipping
- Translation
- 2D rotation
- Zoom in and zoom out

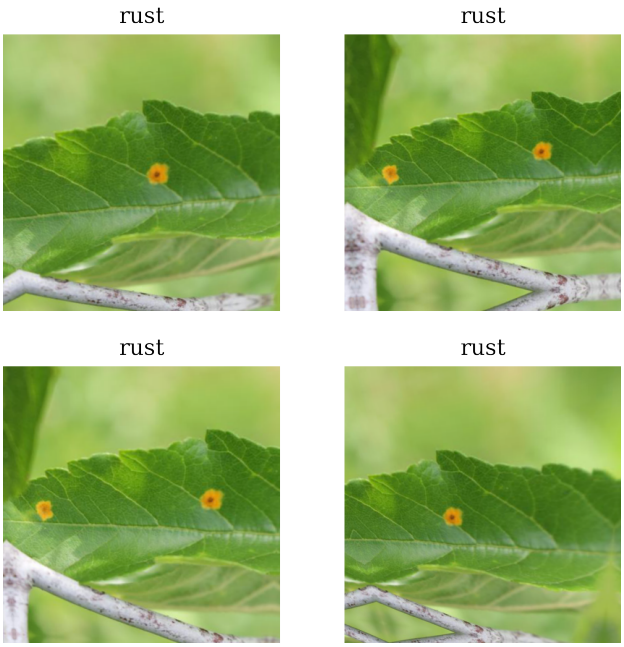

Fig. 3. Samples for the image augmentations applied on an image

Low-precision floating point numbers (FP16) have been used for model weights and gradients to accelerate training [16] [10]. The models are evaluated with the following metrics

implemented in scikit-learn library [17]: accuracy, F1 score with macro and samples averaging.

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{F1} = \frac{2 * TP}{2 * TP + FP + FN}$$

Macro averages F1 scores by class; whereas, samples averages by instance.

The baseline corresponds to a dummy model produces uniform random probability per class.

We benefit from transfer learning by using the models pretrained on ImageNet. Transfer learning means using a subset of weights of a model pretrained on another dataset instead of creating a model from scratch with untrained weights. The final classification layer of each pretrained model is replaced with a new fully connected layer to make them applicable to our dataset. They're trained by fine tuning the classification head for 5 epochs while the parameters of the backbone are frozen; and then, unfreezing the whole model and training until the validation loss saturates.

The experiment results are given in Table I. Aug. stands for image augmentation and its level. BCE corresponds to binary cross entropy loss; WBCE corresponds to binary cross entropy loss with class weights; FL corresponds to focal loss.

As it's seen, ResNet18 architecture with light image augmentation achieves the best performance.

III. Conclusion

It can be concluded that deep and complex models with more trainable parameters don't always achieve the best performance. Neither focal loss nor weighting binary cross entropy loss make any improvement on the model performance. This might be due to the fact that the class imbalance is not very high in the dataset. It can be stated that the model benefits from image augmentation as it generates more samples in various angles and lighting conditions. However, when the augmentation is too heavy, it may distort the images such that important features are lost. Hence, the model performance degrades.

TABLE I
EXPERIMENT RESULTS

Arch.	#param [M]	Img size	Aug.	Loss	Acc.	F1(macro)	F1(samples)
Baseline (uni.random)	0	NA	NA	NA	0.1672	0.277	0.270
ResNet18	11.7	384	None	BCE	0.963	0.888	0.675
ResNet18	11.7	384	1	BCE	**0.966**	**0.895**	**0.690**
ResNet18	11.7	384	2	BCE	0.963	0.889	0.683
ResNet18	11.7	384	1	FL	0.961	0.874	0.671
ResNet18	11.7	384	1	WBCE	0.936	0.792	0.587
ResNet34	21.8	384	2	BCE	0.965	0.891	0.681
EfficientNet-B2	9.2	384	1	BCE	0.957	0.865	0.664
EfficientNet-B3	12.3	512	2	FL	0.960	0.873	0.657
XSE-ResNext50	27.7	384	2	BCE	0.895	0.658	0.443

REFERENCES

[1] R. Thapa, K. Zhang, N. Snavely, S. Belongie, and A. Khan, "The plant pathology challenge 2020 data set to classify foliar disease of apples," *Applications in Plant Sciences*, vol. 8, no. 9, p. e11390, 2020. [Online]. Available: https://bsapubs.onlinelibrary.wiley.com/doi/abs/10.1002/aps3.11390

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 770–778. [Online]. Available: https://doi.org/10.1109/CVPR.2016.90

[3] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 6105–6114. [Online]. Available: https://proceedings.mlr.press/v97/tan19a.html

[4] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[5] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826.

[6] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, ser. AAAI'17. AAAI Press, 2017, pp. 4278–4284.

[7] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," 2017, cite arxiv:1704.04861. [Online]. Available: http://arxiv.org/abs/1704.04861

[8] A. Krizhevsky, "Learning multiple layers of features from tiny images," *Technical Report*, pp. 32–33, 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

[9] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *Int. J. Comput. Vision*, vol. 115, no. 3, pp. 211–252, dec 2015. [Online]. Available: https://doi.org/10.1007/s11263-015-0816-y

[10] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of tricks for image classification with convolutional neural networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 558–567.

[11] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[12] J. Howard and S. Gugger, "Fastai: A layered api for deep learning," *Information*, vol. 11, no. 2, 2020. [Online]. Available: https://www.mdpi.com/2078-2489/11/2/108

[13] L. N. Smith, "No more pesky learning rate guessing games," *CoRR*, vol. abs/1506.01186, 2015. [Online]. Available: http://arxiv.org/abs/1506.01186

[14] ——, "A disciplined approach to neural network hyper-parameters: Part 1 - learning rate, batch size, momentum, and weight decay," *CoRR*, vol. abs/1803.09820, 2018. [Online]. Available: http://arxiv.org/abs/1803.09820

[15] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, p. 60, 2019. [Online]. Available: https://doi.org/10.1186/s40537-019-0197-0

[16] P. Micikevicius, S. Narang, J. Alben, G. Diamos, E. Elsen, D. Garcia, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, and H. Wu, "Mixed precision training," 2018.

[17] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, 2013, pp. 108–122.